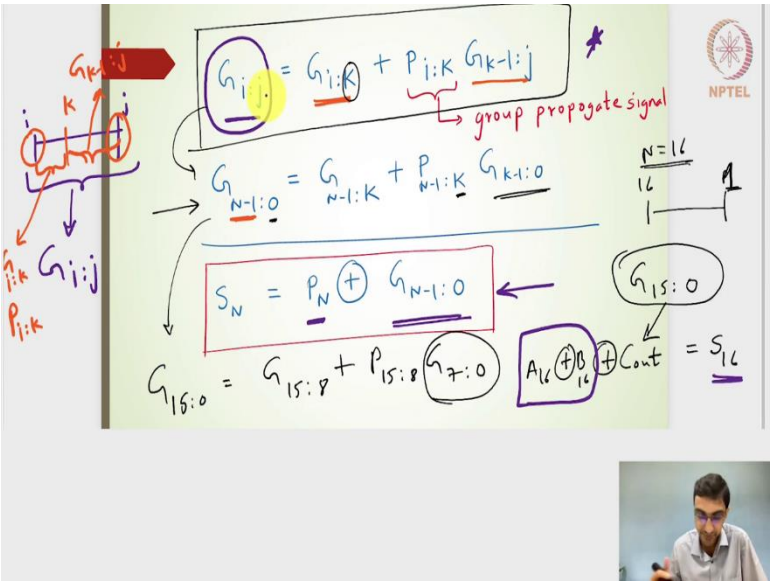
Design and Analysis of VLSI Subsystems
Dr. Madhav Rao
Department of Electronics and Communication Engineering
International Institute of Information Technology, Bangalore

Datapath subsystems – PG Architecture
Lecture - 87
PG architecture - Part1

Hello students, welcome to this lecture on the PG architecture PG represents a propagate and generate signals and then based on that how do we design the adder subsystem blocks.

(Refer Slide Time: 00:30)



Let us move ahead, let me also get my pointer, this is what we had already seen earlier. The group generate signal i and j being, where i is the MSB bit and j is the LSB bit. If I have a group of the bits and then this group of the bits, I can actually generate a group a generate signal. The whole group of bits we should be able to generate a G signal or a generate signal which, I think we can also use this particular expression to generate this i to j bit generate signals.

In that sense if I can actually use this particular expression, what this expression says is if i to j is the group of the bits that we want to extract the generate signals. Then somewhere here we can define a k bit. The k will be in between the MSB and the LSB bit and it is nothing but generating a generate, signal extracting a generate signal from i:k and then extracting the propagate group propagate signal i to k.

Then we can have this particular group, this is $G_{k-1:j}$ extracting the other group. If I have a demarcation in the form of a k bit, then I need to extract the higher MSB bit groups generate signals and then the propagate signals. The group generate and group propagate signals and also on the lower level of the bits from k-1:j, we need to extract the group generate k-1:j bits generate signals.

This is how it is been expressed and we need to utilize this to design our different adder subsystem blocks. If I use, i = n - 1 and j = 0, what it means is if I have 1 to 16 bits, somewhere here I will write it down, if I have the 16th-bit and starting from the 1-bit.

If I want to develop an adder system 16 to 1-bit, then where n is equal to 16, I can then utilize $G_{15:0}$, I need to extract $G_{15:0}$. This will be considered as the carry out for the 16th bit, this will be considered as a carry out for the 16th bit and if I do an XOR with that of A_{16} XOR B_{16} XOR Cout I should be able to get this sum 16 bit. That is why we have taken this generate group, generate n-1:0.

Now, to get this group 15:0 group generate 15:0, I can utilize demarcate this group of bits with a k value. Where the k value can lie between 1 and j, inclusive of 1 and j. In that sense I am defining, I am using this particular expression, whatever is considered here utilizing that into N-1:0.

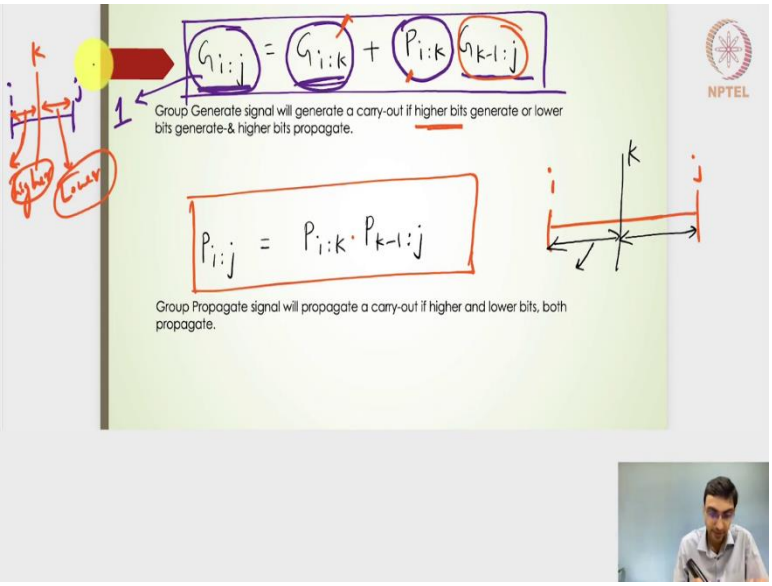
I am using a k in between N-1:0. The N-1 to k + P_{N-1} is to k and with that of the group generate k-1:0. If I use this particular expression, where the group generate signal is always with respect to the 0. Whatever the group I generate $G_{15:0}$, $G_{14:0}$, $G_{13:0}$, if I consider $G_{15:0}$, this k value can be anything like 8. In that sense this k-1 will be 7:0. If I take a simple example where N = 16.

$$G_{15:0} = G_{15:8} + P_{15:8}G_{7:0}$$

So, that I will eventually get $G_{15:0}$ and $G_{15:0}$ is used in the extracting the sum bit of S_{16}. That is what I have mentioned here in this particular expression, S of N which is the sum bit of the nth sum bit is nothing but the propagate of N-bit which is nothing but A and B XOR with that of B and that of XOR with that of $G_{N-1:0}$.

What we are going to use is this particular expression in different forms or rather apply the k value as either the previous value or the different values of k here, and then different values of j here and then try to see how do we design the adder subsystem blocks.

(Refer Slide Time: 06:28)



Again, just to dig deeper into this particular expression, if I look into this particular expression in terms of i j, it is nothing but,

$$G_{i:j} = G_{i:k} + P_{i:k}G_{k-1:j}$$

If $G_{i:j} = 1$, if this is 1, that means $G_{i:k} = 1$ or $P_{i:k} = 1$, but at the same time $G_{k-1:j} = 1$.

What it really means is the group generate signal, the group generate signal for i j will generate a carry out. If this has to be 1, because i j is nothing but considered as a carryout if the higher bits i k considered to be the higher bits. If I consider i to j and then, let me say I will demarcate it with the k bit in between.

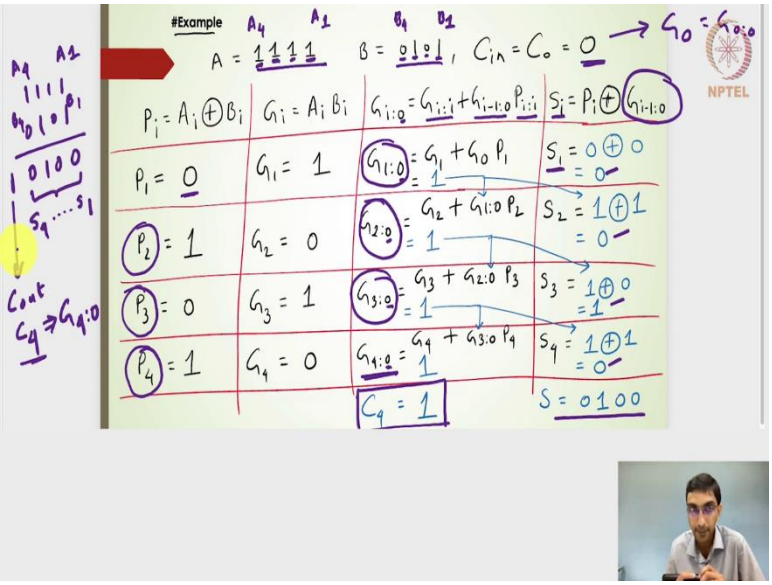
This particular bit are called the higher bits and then this particular bits are called the lower bits, that is what I am saying. The higher bits is going to generate or this is going to generate a 1 here or the lower bits is going to generate and higher bits are going to propagate. This is going to either generate or this will generate, but this is going to be simultaneously going to propagate.

This i k is nothing but the higher bits propagate group propagate signal. When this is propagating, this the lower bits has to be generated or this higher level of bits should be able to generate a carry out. Then only I will have $G_{i:j} = 1$. For propagate signal by definition it is

$$P_{i:j} = P_{i:k}P_{k-1:j}$$

If I have i:j as a group of bits and then if I demarcate with a k bit, if I choose a k value in between i and j then for the group propagate, for i to j it will be 1 if i to k the higher level of bits has to propagate and at the same time the lower level of the bits also has to propagate. The group propagate signal will propagate a carry out if higher and lower bits both propagate. Hope this is clear, moving forward.

(Refer Slide Time: 09:20)



Let us take an example here of A and B with the carry input as 0. The A and B is nothing but A is all 1s and B is 0 1 0 1. If I actually do the addition here of 1, 1, 1, 1 and then 0 1 0 1.

If I actually do an addition here, it will turn out to be like this whereas this are all the sum bits. Starting from S_4 to S_1, S_4 is 0, S_1 is 0, S_3 is 1, S_2 is 0 and this will be our carry out or we can say that it is nothing but C_4.

The carry in is Cin or C_0 = 0. I have not taken any carry in, C_4 is our carryout. Now, this is what we know, let us try to utilize that group generate and then group propagate signals

or rather the group generate signals and then try to see whether it matches with our expected sum and the carryout values.

The C_4 is actually in terms of the group generate signal, I can say that it is nothing but $G_{4:0}$ and sum 4, sum 1, sum 2, sum 3 I can actually represent it in the form of the propagate signals for that particular bit level and XOR with that of the previous group generate signals.

If I consider this to be A_4 to A_1 and then this one as B_4 to B_1, then I should be able to generate what is P_4, P_1, P_2, P_3. Similarly, G_4, G_3, G_2, G_1 and then get those values of the group generate signals and then extract the sum and then the Cout expression.

Let us try to do that particular thing and then see whether it validates to our expected results. I have written here the bit level propagate signal for the 4-bits here is nothing but A_i XOR B_i and bit level generate signal is A_i B_i and operation. In that sense if I can redo this or rather, I will say that this is a 4-bit and then this is a 1-bit.

Then this is B_4 bit and then B_1 bit. We know that this is the LSB side and this is the LSB side and then this is the MSB side, this is the MSB side. What should be P_1, then P_1 should be nothing but A_1 XOR B_1. The A_1 XOR B_1 will be 0, P_2 is A_1 or rather A_2 XOR B_2 which will be 1.

The P_3 will be 0, P_4 will be 1 and 0 it will be 1, bitwise generate signal will be nothing but the AND operation. The AND operation will give me 1 and 1, will be 1 here G_2 will be 1 and 0 will be 0, G_3 will be 1 and 1 will be 1 G_4 will be 1 and 0 will be 0. This bitwise generate and propagate signals are done.

Let us try to extract the group wise, the way we will do the group wise is $G_{i:0}$. Our base bit is always 0 here, if I have the base bit to be always 0 that is nothing but the carry out of the previous addition bits and that will be utilized, that will be utilized for the present bit addition.

If I want to find out the S_4 bit, the S_4 bit will be nothing but the XOR operation or rather the P_i and XOR with that of the group generate of i-1:0. The Si is nothing but P_i XOR with that of $G_{i-1:0}$, that is why I need this G_{i-1}.

What we really need is $G_{0:0}$ and then $G_{1:0}$, that I can calculate S_2 $G_{2:0}$ so that I can calculate S_3, $G_{3:0}$. So, that I can calculate S4, $G_{4:0}$ is nothing but the carry out of this 4-bit addition which will be C_4. In that sense what I need is $G_{1:0}$ is nothing but if I consider k as nothing but i here because k is inclusive of i and j that is what I have told.

If I consider k = i, this will be $G_{i:i} + G_{i-1}P_{i:i}$, i:i will become instead of a group wise because it is the same level of the bit. It becomes an individual group generate bit individual generate bit $P_{i:i}$ will also become an individual propagate bit, i-1:0, i :1 or rather i-1:0, this will be the group generate from i-1:0.

If I consider i is to be 1 here, then this will become $G_{1:1}$ which will be nothing but G_1, this will become 0:0 which will be nothing but G_0 and this will become $P_{1:1}$, which will be nothing but the bit level propagate P_1 signal. The G_1 we know it is nothing but 1 or with that of G_0, G_0 is considered the carry input is 0 here. This is nothing but G_0 signal or also we can write it as $G_{0:0}$.

That will be nothing but 0 and $P_{1:0}$, it will give me the value of 1 here, $G_{2:0}$ will be nothing but I will consider i to be 2 now. The G_2 + $G_{1:0}$ and with that of P_2 $G_{1:0}$ is calculated as 1 here. This 1 will go here, $P_{2:1}$, G_2 will be 0. It will be 0 or that of 1 and 1 which will be 1 here.

The $G_{3:0}$, will be nothing but G_3 + $G_{2:0}$ and then P3, $G_{2:0}$ = 1, $P_{3:0}$, G_3 is anyways 1, this value will be 1. The $G_{4:0}$ will be nothing but G_4 which will be 0 plus this 1 will come G_3 is to 0 will be 1 and P_4 is nothing but 1, I will get a value of 1.

Note that all my group generate signals starting from $G_{1:0}$ to $G_{4:0}$ is actually 1. The $G_{4:0}$ is 1 represents that my carry out of the 4-bit addition is actually 1. Now, use this particular group generate signals to extract our sum bits. The S_1 is nothing but $G_{0:0}$, XOR with that of P of 1.

The P of 1 is 0, XOR 0 will be 0, S_2 will be this particular element, XOR with that of the P_2 signal. It will be S_2 will be nothing but 1, XOR 1 will be 0. The S_3 will be P_3 XOR with that of $G_{2:0}$, that will be 1 XOR 0 it will be 1.

The S_4 will be P_4 XOR $G_{3:0}$ which will be 1 XOR 1 it will be 0. My sum bits are actually 0, 0, 1, 0. The 0, 0, 1, 0, and my carry output is 1, that is what I had got, I had got 0, 1, 0, 0 and then they carry out as 1.

If I use this particular group generate signals, it will indeed give me the same answer as what I used to do for the manual edition. Using this particular group generate and then group propagate or individual generate bit or the individual propagate bits, we will now try to construct the adder subsystem block.